REMARKS

Claims 1-20 are pending. Applicants respectfully traverse and request reconsideration and withdrawal of the rejections for the following reasons.

I. Summary of the Examiner's Objections/Rejections

Claims 1-10 and 12-19 remain rejected under 35 U.S.C. §103(a) over U.S. Patent Number 6,446,193 to Alidina et al. (Alidina), in view of U.S. Patent No. 5,896,517 to Wilson (Wilson). Claims 11 and 20 are rejected under 35 U.S.C. §103(a) over Alidina, in view of Wilson and in further view of U.S. Patent Number 5,673,377 to Berkaloff. Claim 16 is objected to due to an informality.

II. Applicant's Response to the Examiner's Rejections

The Applicants traverse the aforementioned claim rejections for at least the reasons set forth in greater detail below.

A. Objection to Claim 16.

The Applicants' attorney thanks the Examiner for the suggestion with respect to Claim 16. Claim 16 is amended to correct a typographical error.

B. 35 U.S.C. §103(a); Claims 1-10 and 12-19.

Claims 1-10 and 12-19 are rejected under 35 U.S.C. §103(a) over Alidina, in view of Wilson.

Alidina

Alidina is directed to reducing instruction cycles in a digital signal processor by processing two different register parts in a single processor cycle instead of two. (Alidina, Col. 2, lines 42-56). Alidina teaches that repetitive operations can be carried out in parallel, so as to reduce the number of instructions and, enhance the MIPs number by freeing up valuable instruction time for other operations. (Alidina, ¶ 3, lines 49-52). (Emphasis added). Since

multi-threading does not reduce the number of instructions, nor enhance MIPs, Alidina teaches away from multi-threading. Alidina further teachers that the data arithmetic unit comprises a power efficient dual-MAC parallel pipelined structure particularly optimized for wireless and speech synthesis applications. (Alidina ¶4, lines 23-25). (Emphasis added). The Office Action acknowledges that Alidina does not teach multi-threading and having registers and operands that correspond to each individual thread.

Wilson

Wilson is directed to a method for improving the performance of a computer system by performing useful work in parallel with long-latency main memory accesses. (Wilson ¶3, lines 14-21, ¶3, lines 4-8). (Emphasis added). Wilson throughout the specification repeatedly describes performing useful work in parallel. (Wilson, Abstract ¶2, line 6; line 3; ¶2, lines 63-65; ¶3, lines 7, 21, 36; ¶13 line 3; ¶15, line 14; ¶16 line 23; ¶18, line 34; claim 1, claim 17). Because of the high overhead costs, and because such process-swapping is always unexpected, "It is desirable to avoid the overhead costs of process-swapping," i.e., the multi-threaded processor scheme. (Wilson, Col. 2, line 66–Col. 3 line 3). The overhead of multi-threaded process swapping (such as needing data from a disk due to a cache miss) can be quite expensive since there may be many registers to save and restore. (Wilson, Col. 2, lines 34-65). Instructions are added at the program writing stage by the programmer or by software tools, such as a compiler, to pre-fetch data from main memory in order to avoid the overhead of multi-threaded process swapping because of blocking, such as a cache miss. (Wilson, Col. 3, lines 14-28). (Emphasis added)

The combination of Alidina and Wilson, to the extent Alidina and Wilson may be combined, would teach reducing instruction cycles in a digital signal processor by processing in

parallel two different register parts in a single processor cycle, instead of two swapped threaded processor cycles, through adding instructions at the program writing stage by the programmer or by software tools, such as a compiler, to pre-fetch data from main memory in order to avoid the overhead of multi-threaded process swapping.

NEITHER ALIDINA NOR WILSON TEACHES AT LEAST "WHEREIN A SELECTED ACCUMULATION REGISTER THAT CORRESPONDS TO THE SELECTED THREAD STORES THE FIRST OPERATION RESULT CORRESPONDING TO THE SELECTED THREAD"

To establish a *prima facie* case of obviousness, each and every element arranged, as required by the claims, must be taught or suggested in the prior art. MPEP 2143.03.

The Office Action dated 1/13/04, at page 15 ¶ 33 states:

> the sentence being referred to in Wilson has been read out of context." Read in context with the rest of the cited section, a person of ordinary skill in the art would recognize that Wilson is teaching that, in a multi-threaded system, to increase efficiency, it is desirable to minimize the overhead costs. This does not teach away from multi-threading. The statements in Wilson, when read in context, teach how to increase the efficiency of multi-threaded systems.

However, the Office Action fails to show wherein a selected accumulation register that corresponds to the selected thread stores the first operation result corresponding to the selected thread or otherwise show the language in Wilson teaching a "multi-threaded system." Contrary to the assertion in the Office Action, Wilson in the Summary of the Invention explicitly describes a preferred embodiment of the invention "as avoiding the overhead of process-swapping." (Wilson ¶3, lines 25-28). (Emphasis added).

Even though the Office Action fails to show how the sentence is read out of context, the first paragraph of the summary of the invention states the following: This mechanism makes it possible to improve the performance of the computer system through the effective use of added concurrency while avoiding the overhead of process-swapping. (Wilson, ¶3, lines 25-28,

emphasis added). Consequently, Wilson explicitly avoids process-swapping with respect to the objects in a preferred embodiment of the present invention. Further, with respect to the Background of the Invention, Wilson explicitly states "it is the primary object of the present invention . . . to do useful work in parallel with long latency main memory accesses." (Wilson, ¶3, lines 4-5). (Emphasis added) Therefore, the assertion in the Office Action in ¶35 that the "lines relied upon" in the Background of the Invention did not include anything dealing with the invention of Wilson nor its objectives is contradicted by the explicit teachings of Wilson at Column 3, lines 4-8. Therefore, in view of the explicit teachings of Wilson namely, "avoiding the overhead of process-swapping" a person of ordinary skill in the art would recognize that Wilson is teaching away from a multi-threaded system since process-swapping would increase overhead and, the associated increased costs, something Wilson explicitly seeks to avoid.

The Office Action fails to establish a *prima facie* case of obviousness because the Office Action fails to show how each and every element in the claims is taught by the references. Among other things, the Office Action fails to show where the combination of Alidina and Wilson teach "wherein a selected accumulation register that corresponds to the selected thread stores the first operation result corresponding to the selected thread." Further, Alidina is directed to processing two different register parts in a single processor cycle for speech coding and, as a result, is directed to a problem different than the claims. (Alidina, Col. 2, lines 42-56). Further, Wilson fails to teach, among other things, a multi-threaded system as claimed. As previously stated, Wilson instead teaches, "it is desirable to avoid the overhead costs of processing swapping," i.e., multi-threaded context switching, because multi-threaded switching can be quite expensive, since there may be many registers to save and restore. (Wilson, Col. 3, lines 26-29). Further, Wilson, as cited, fails to teach "wherein a selected accumulation register that

corresponds to the selected thread stores the first operation result corresponding to the selected thread" because, in contrast to the claims, Wilson teaches away from multi-threaded processing. *Id.* Further yet, Wilson teaches away from storing a thread in an accumulation register because Wilson teaches "multi-threaded context switching can be quite expensive since there may be many registers to save and restore." Therefore, the combination of Alidina, in view of Wilson, as cited, fails to teach, and further teaches away from, among other things, "wherein a selected accumulation register that corresponds to the selected thread stores the first operation result corresponding to the selected thread" as arranged in the claims. Consequently, the Office Action fails to establish a *prima facie* case of obviousness.

There is no motivation to combine Alidina with Wilson because Wilson teaches away from the claims since, as previously stated, Wilson teaches avoiding multi-threaded context switching to avoid the expense of "many registers to save and restore." (Wilson, Col. 2, line 66–Col. 3, line 3).[1] As previously stated, Wilson teaches "with a multi-threaded processor, context switching can be quite expensive since there may be many registers to save and restore" (Wilson, ¶2, lines 63-65) and, therefore, "It is most desirable to avoid the overhead costs of process-swapping." The language relied upon by the Office Action regarding multi-threading, actually teaches the avoidance of multi-threading and because Wilson teaches "it is most desirable to avoid the overhead costs of process-swapping. Therefore, Wilson teaches away from multi-threaded process-swapping.

The Office Action at ¶ 33 asserts that, "read in the context with the rest of the cited section, a person of ordinary in the art would recognize the Wilson is teaching that, in a multi-threaded system, to increase efficiency, it is desirable to minimize the overhead costs" and this

does not teach away from multi-threading." However, this assertion is unsupported since the Office Action provides no citation in Wilson or in Alidina to support this assertion. Further, this assertion is explicitly contrary to the teachings of Wilson. The very portion of Wilson cited in the Office Action to provide motivation to combine Alidina with the teachings of Wilson (1) teaches coding at the program writing stage to pre-fetch data, (2) teaches away from multi-threaded context switching, and (3) teaches away from saving and restoring registers and therefore explicitly teaches away from the claims. As a result, there is no motivation to combine the references since Wilson teaches against the claims. Consequently, the Office Action fails to establish a *prima facie* case of obviousness.

As previously stated, to the extent Alidina and Wilson may be combined, such a combination would teach reducing instruction cycles in a digital signal processor by processing two different register parts in parallel in a single processor cycle, instead of two swapped threaded processor cycles, by adding instructions at the program writing stage, by the programmer or by software tools, such as a compiler, to pre-fetch data from main memory in order to avoid the overhead of multi-threaded process swapping. As a result, the combination of Alidina and Wilson is directed to solving a completely different problem than the claims. Therefore, the combination of Alidina in view of Wilson, as cited, teaches away from the claims, namely process swapping, and fails to teach each and every element as arranged in the claims. Consequently, the Office Action fails to establish a *prima facie* case of obviousness.

[1] A prior art reference must be considered in its entirety, i.e., as a whole, including portions that would lead away from the claimed invention. (*W.L. Gore & Associates, Inc. v. Garlock, Inc.*, 721 F.2d 1540, 220 USPQ 303 (Fed. Cir. 1983), cert. denied, 469 U.S. 851 (1984) MPEP 2141.02).

THE MODIFICATION OF ALIDINA, AS SUGGESTED BY WILSON, WOULD CHANGE THE PRINCIPLE OF OPERATION OF THE CLAIMED INVENTION, AND THEREFORE THE OFFICE ACTION FAILS TO PROVIDE ANY MOTIVATION TO COMBINE THE REFERENCES

The modification of Alidina, in view of the cited suggestion of Wilson, would change the principle of operation of the claimed invention because the modification suggested by Wilson, as previously stated, rather than teaching multi-threaded context switching, teaches "it is desirable to avoid the overhead costs of process-swapping." Instead of multi-threaded processing as claimed, Wilson teaches coding at the program writing stage to pre-fetch data. Further, the proposed combination teaches multi-threaded context switching can be quite expensive, since there may be many registers to save and restore. Therefore, the modification suggested by Wilson would change the principle of operation of the claimed invention because Wilson (1) teaches coding at the program writing stage to pre-fetch data, (2) teaches against multi-threaded context switching, and (3) teaches against saving and restoring registers.[2] Consequently, for at least these reasons, there is no motivation to combine the references and, therefore, the Office Action fails to establish a *prima facie* case of obviousness.

NEITHER ALIDINA NOR WILSON TEACHES AT LEAST "A SELECTION BLOCK OPERABLY COUPLED TO THE PLURALITY OF ACCUMULATION REGISTERS AND THE FIRST OPERATION UNIT, WHEREIN THE SELECTION BLOCK SELECTS THE SECOND OPERAND PROVIDED TO THE FIRST OPERATION UNIT FROM A SET OF POTENTIAL OPERANDS, WHEREIN THE SET OF POTENTIAL OPERANDS INCLUDES CONTENTS OF EACH ACCUMULATION REGISTER OF THE PLURALITY OF ACCUMULATION REGISTERS"

The cited portion of Alidina at ¶5, lines 63-66, which states "the modified results from the respective saturators SAT1, SAT2 and SAT3 are then fed through a split multi-plexer SMUX to a register array 50 comprising eight 40-bit accumulators a0 through a7 is limited to modifying

[2] If the proposed modification or combination of the prior art would change the principle of operation of the prior art invention being modified, then the teachings of the references are not sufficient to render the claims *prima facie* obvious. In re Ratti, 270 F.2d 810, 123 USPQ 349 (CCPA 1959). See MPEP 2143.01.

results from saturators SAT1, SAT2 and SAT3 are fed through split multi-plexer SMUX to a register array 50 rather than "wherein a selected accumulation register that corresponds to the selected thread stores the first operation result corresponding to the selected thread."

As previously stated, the Office Action acknowledges that Alidina does not describe "multi-threading and having registers and operands which correspond to each individual thread." Accordingly, Alidina also fails to teach "wherein a selected accumulation register that corresponds to the selected thread stores the first operation result corresponding to the selected thread.

According to the Office Action in ¶35, "as for the path, which passes through other blocks, the claim language states 'comprising' which is open. As long as the elements recited in the claim are taught in the reference, the extraneous elements do no matter." However, in order to show that Alidina teaches "wherein the selection block selects the second operand provided to the first operation unit from a set of potential operands", the Office Action must show where Alidina teaches all the limitations as arranged in the claims, including, among other things, this claimed routing. Therefore, since the Office Action fails to show where Alidina teaches this specific routing, namely "wherein the selection block selects the second operand provided to the first operation unit from a set of potential operands", the Office Action fails to show where either Alidina or Wilson in combination teaches all of the elements as arranged in the claim. Nevertheless, the path, as asserted, passes through other blocks and, therefore some teaching must be shown for routing the signal through only those blocks out of the many blocks in order to form the specific path. Fig. 3 fails to show the specific routing path and, further, fails to show the required control signals to establish any such path, as asserted in the Office Action. As a result, the combination of Alidina and Wilson fails to teach "a selection block operably coupled

to the plurality of accumulation registers and the first operation unit, wherein the selection block selects the second operand provided to the first operation unit from a set of potential operands, wherein the set of potential operands includes contents of each accumulation register of the plurality of accumulation registers." (Alidina, Col. 4, lines 57-59).

Fig. 3 teaches performing arithmetic functions rather than a multi-thread process and, therefore, solves a different problem than the instant application. Consequently, the extra functional blocks for performing the arithmetic functions beyond those needed for performing process swapping in the multi-thread environment would add unnecessary expense and complexity and would hinder performance in a multi-thread environment because of the unnecessary hardware and processing. Accordingly, the complex arithmetic unit taught by Alidina actually teaches away from the claimed invention because the added complexity and the resultant reduction of speed and performance would not be suitable for performing process swapping in a multi-thread environment. Furthermore, modifying the arithmetic logic unit in Alidina, as suggested in the Office Action, to perform process swapping in a multi-thread environment, would require elimination of the arithmetic functions, thus rendering Alidina unsatisfactory for its intended purpose as an arithmetic unit, since the arithmetic unit would need to be removed to achieve the performance requirements for process swapping.[3] Nevertheless, the Office Action at ¶37 states that the examiner "is unsure how this argument is related to the claim language and related to the invention." The Applicants would like to point out the distinction between arithmetic logic until performing arithmetic functions and process swapping. As previously stated, since Alidina teaches an arithmetic logic unit, such an arithmetic logic unit performs arithmetic functions, rather than perform multi-threaded processes as previously stated.

The Office Action asserts that the SMUX, as cited in Alidina, is the claimed selection block. However, as shown above, Alidina does not teach a path from the control registers via the SMUX to the first operation unit (no equivalent element in Alidina identified in the Office Action), but rather the XYFB couples SMUX to the register X(32). In Alidina, "the eight accumulators are controlled according to modes defined by preselected mode bits provided by control registers auc0 and auc1 to selectively provide feedback along a feedback path XYFBK to the x-y multiplier registers x(32) and y(32)." (Alidina Col 5. lines 13-17). If the rejection is maintained, and the registers x(32) in Alidina are equated to the first operation unit rather than the accumulation registers, then Alidina fails to teach at least "wherein the operation unit combines the first and second operands to produce a first operation result corresponding to the selected thread," as claimed. Alidina does not teach, and Applicants request a showing of, "wherein the selection block selects the second operand provided to the first operation unit from a set of potential operands, wherein the set of potential operands includes contents of each accumulation register of the plurality of accumulation registers." Therefore, for at least these reasons, the combination of Alidina, in view of Wilson, as cited, fails to teach each and every element as arranged in the claims. Consequently, the Office Action fails to establish a *prima facie* case of obviousness.

[3] If the proposed modification would render the prior art invention being modified unsatisfactory for its intended purpose, then there is no suggestion or motivation to make the proposed modification. *In re Gordon*, 733 F.2d 200, 221 USPQ 1125 (Fed. Cir. 1984), MPEP 2143.02.

NEITHER ALIDINA NOR WILSON TEACHES AT LEAST "A FIRST OPERATION UNIT OPERABLY COUPLED TO RECEIVE A FIRST OPERAND AND A SECOND OPERAND CORRESPONDING TO AN OPERATION CODE ISSUED BY A SELECTED THREAD OF THE PLURALITY OF THREADS, WHEREIN THE OPERATION UNIT COMBINES THE FIRST AND SECOND OPERANDS TO PRODUCE A FIRST OPERATION RESULT CORRESPONDING TO THE SELECTED THREAD"

The Office Action cites Alidina (Alidina, Abstract, lines 1-4; ¶2, lines 46-48; ¶4-5, lines 26-7; and Figure 3) for teaching "a first operation unit operably coupled to receive a first operand and a second operand corresponding to an operation code issued by a selected thread of the plurality of threads." However, as stated in the previous response, this general reference to over 70 lines in a reference without showing equivalent claim elements does not make apparent, nor explain with specificity, how each claim is rejected. MPEP 706, CFR 1.104(c)(2). For example, Applicants are unable to find any reference in Alidina, as cited, "corresponding to an operation code issued by a selected thread of the plurality of threads," as claimed. As previously stated, Applicants cannot find where Alidina teaches multi-threaded processing.

According to the Office Action at ¶38, "Wilson was relied upon to teach multi-threading." As previously stated, Wilson explicitly teaches away from a multi-threaded accumulation circuit, since Wilson teaches "it is desirable to avoid the overhead costs of processing swapping," and multi-threaded context switching can be quite expensive, since there may be many registers to save and restore. Despite the explicit language in Wilson that teaches away from multi-threaded processing, the Examiner continues to ignore these explicit teachings and persists instead to assert that "Wilson was relied to teach multi-threading." (Office Action ¶38). To the extent Alidina and Wilson may be combined, such a combination would teach reducing instruction cycles in a digital signal processor by processing two different register parts in parallel in a single processor cycle instead of two swapped threaded processor cycles. Consequently, among other things, the combination of Alidina, in view of Wilson, as cited, fails

to teach and Applicants request a showing of "a first operation unit operably coupled to receive a first operand and a second operand corresponding to an operation code issued by a selected thread of the plurality of threads." Therefore, the combination of Alidina, in view of Wilson, as cited, fails to teach each and every element as arranged in the claims. Consequently, the Office Action fails to establish a *prima facie* case of obviousness.

The Office Action fails to indicate, and Applicants are unable to find, any reference to where Alidina or Wilson, in combination or individually as cited, teaches "wherein the operation unit combines the first and second operands to produce a first operation result corresponding to the selected thread." Since the combination of Alidina and Wilson fails to describe multi-threading and process swapping, as previously described, the combination of Alidina and Wilson fail to describe a "selected thread." Consequently, the combination of Alidina and Wilson teaches against "wherein the operation unit combines the first and second operands to produce a first operation result corresponding to the selected thread." Again, if the Examiner maintains this rejection, Applicants again respectfully request a showing in Alidina and Wilson of each and every element arranged as claimed.

C. 35 U.S.C. §103(a); Claims 11 and 20.

Claims 11 and 20 are rejected under 35 U.S.C. §103(a) over Alidina in view of Wilson, and in further view of U.S. Patent Number 5,673,377 to Berkaloff. The Office Action acknowledges that Alidina does not teach multi-threading and having registers and operands that correspond to each individual thread. The Office Action acknowledges that Alidina does not teach wherein the first register section accumulates diffuse color information corresponding to graphics primitives, and wherein the second register section accumulates specular color information corresponding to the graphics and primitives. Further, the motivation provided in the Office Action to combine the references, "because it is needed in the calculations to create

effective images," provides no basis for the meaning of "effective images," and further uses circular reasoning, and therefore fails to establish motivation to combine the references. Claims 11 and 20 add additional novel and nonobvious subject matter, and are also allowable at least for the above reasons and as depending from an allowable base claim.

As to claim 2, the Office Action cites Alidina as teaching a control block, as claimed. However, this portion describes bus-accessible control registers rather than "a control block operably coupled to the selection block and the plurality of accumulation registers, wherein the control block receives information based on the operation code and generates control information provided to the plurality of accumulation registers and the selection block, wherein the control information provided to the plurality of accumulation registers causes the selected accumulation register to store the result corresponding to the selected thread when the operation code corresponds to an accumulate operation." The Office Action fails to show how Alidina teaches that the bus-accessible control registers are equivalent to the control block as claimed and, therefore, fails to establish a *prima facie* case of obviousness. Applicants at least reassert that the references do not teach each and every element, as arranged in the claims with respect to claim 1. Claim 2 adds additional novel and nonobvious subject matter for at least these reasons and is also allowable, at least, as depending from an allowable base claim.

As to claims 3, 4, 5, 6, 8, 9, 10, 12, 13, 14, 15, 16, 18, and 19, Applicants at least reassert the above reasons. Each claim adds additional novel and nonobvious subject matter and is at least as depending from an allowable base claim. Further, the combined references do not teach each and every element as arranged in the claims.

As to claims 7 and 17, the cited portion of Wilson, rather than teaching an arbitration module, as asserted in the Office Action, teaches against arbitration in a multi-threaded processor

since Wilson teaches against multi-threaded processing, as previously stated. Accordingly, the assertion that it is inherent, and that there must be a unit as claimed, is improper. A supporting reference is respectfully requested if the rejection is maintained. The Office Action acknowledges that Alidina does not teach multi-threading and having registers and operands that correspond to each individual thread. Further, the Office Action on p.12, ¶26 acknowledges that Alidina fails to teach an arbitration module that receives command codes corresponding to a plurality of threads, wherein at least a portion of the command codes correspond to a plurality of threads, wherein at least a portion of the command codes correspond to multiply and accumulate operations and wherein the arbitration module determines an order of execution of the command codes. Applicants also at least reassert the above statements, *inter alia*, and that the references do not teach each and every element as arranged in the claims with respect to claim 1. Claims 7 and 17 add additional novel and nonobvious subject matter, and are also allowable at least as depending from an allowable base claim.

Applicants respectfully submit that the claims are in condition for allowance, and respectfully request that a timely Notice of Allowance be issued in this case. The Examiner is invited to contact the below-listed attorney if the Examiner believes that a telephone conference will advance the prosecution of this application.

Respectfully submitted,

By _____
Themi Anagnos, Reg. No. 47,388

Dated: March 15, 2004

Vedder, Price, Kaufman & Kammholz, P.C.
222 North LaSalle Street
Chicago, Illinois 60601
PHONE: (312) 609-7970
FAX: (312) 609-5005